Supporting Information: Structural Propensities of Kinase Family Proteins from a Potts Model of Residue Co-Variation

Allan Haldane, William F. Flynn, Peng He, R. S. K. Vijayan, and Ronald M. Levy

DATA PROCESSING

Sequence Alignment

We use HHblits [1] (as suggested in [2]) to search the Uniprot database starting from from the Pfam kinase family seed (PF00069) in two passes. We modified the HHblits code (version 2.0.16) to overcome the alignment size limit of 2^{16}. In the first pass we use options `-M 20 -n 2 -neffmax 1000 -all -e 1e-3 -p 90 -maxfilt 1000000 -B 200000 -Z 100000` to detect kinase sequences with high p-value. We perform a second pass using the first pass as a seed with extra parameters `-n 1 -global -mact 0` to obtain complete kinase domains. We remove any sequences with gaps in the "HRD" or "DFG" triplets, sequences missing the aspartic acid required for $Mg2^+$ binding, or more than 10 gaps, or more than 40 inserts, or with invalid/unknown amino acids, leaving 127113 sequences of length 241.

Phylogenetic Filtering

Our ideal dataset would be a set of independent sequences in evolutionary "thermodynamic equilibrium", however in practice the observed sequences have phylogenetic relationships. As described in [3], we roughly account for phylogeny by reweighting sequences by their frequency in the MSA. That is, we assign a weight $w = 1/n$ to each sequence, where n is the number of sequences in the alignment with greater than 40% sequence identity to it (in the 21 letter alphabet). We compute the dataset bivariate marginals with these weights. This leaves an "effective" number of sequences $N_{\text{eff}} = \sum w$ of 8149. We then trim the first 5 and last 61 positions from the alignment which contain variable secondary structures, leaving 175 positions.

Alphabet Reduction

The computational cost of our inference procedure (described below) is significant, and for a typical protein with $L \approx 200$ positions and $q = 21$ residue types there are $\binom{L}{2}q^2 \sim 10^7$ parameters to fit based on an equal number of observed marginals. We mitigate this issue by reducing the alphabet size. We randomly choose a position in the alignment and find the pair of letters which, when treated as identical, would minimize the root mean square difference between the Mutual Information (MI) scores for all $\binom{L}{2}$ position pairs in the reduced alphabet and full 21 letter alphabet. The MI score is calculated as $\text{MI}^{i,j} = \sum_{\alpha,\beta} f^{ij}_{\alpha\beta} \log(f^{ij}_{\alpha\beta}/f^i_\alpha f^j_\beta)$. This is repeated until all positions have been reduced by one letter, which is repeated until all positions have been reduced to 8 letters. This is robust to the random realization and allows the alphabet reduction to differ across positions.

Unlike amino acid reduction schemes based on physiochemical properties, this method is designed to preserve the correlation structure of the MSA. It also reduces the effect of sampling error in each measured marginal. For the kinase MSA we find that reduction to 8 letters is a suitable compromise between reducing the problem size and preserving the sequence correlations (figure S1B), and captures almost all the sequence variation: Kinase sequences which have 27% average pairwise identity with 21 letters still only have 31% identity with 8 (figure S1A). Further justifying this choice, the mean effective number of amino acids at each position of our raw dataset is 8.9, computed by exponentiating the site-entropy as $q_i^{\text{eff}} = e^{-\sum_\alpha f^i_\alpha \log f^i_\alpha}$.

Finite Sample Size Correction

We add a small pseudocount to the bivariate marginals computed from the dataset as a finite size correction. Adding a small flat pseudocount (e.g. of $1/N$) would be equivalent to adding a small number of completely random sequences to the original sequence dataset. Instead we add a pseudocount to mimic a dataset composed the original sequences but with a small per-position chance μ of mutating to a random residue. With this strategy the pseudocounted bivariate marginals are given by

$$(f^{ij}_{\alpha\beta})_{\text{pc}} = (1-\mu)^2 f^{ij}_{\alpha\beta} + \frac{(1-\mu)\mu}{q}(f^i_\alpha + f^j_\beta) + \frac{\mu^2}{q^2} \quad (1)$$

We choose a pseudocount parameter of roughly $\mu = 1/N_{\text{eff}}$.

PDB datasets

We collect 2869 kinase structures from the PDB database by searching for Uniprot IDs corresponding to Mouse and Human kinases according to the Uniprot database. We align their sequences to the Uniprot dataset, and further filter on the following criteria: We remove any sequences with more than 32 gaps after alignment, structures which were crystallized with SH2 domains present (which may crystallize into unusual conformations), structures in which the activation loop is

unnaturally extended due to interactions across the crystal unit cell (e.g. PDB-ID 2WTC), and structures classified as DFG-in but in which the expected β-3 sheet Lys to α-C helix Glu salt bridge distance is more than 5Å . This filtering was performed through cutoffs on relevant residue-residue distances. We use annotation from the KLIFS database. Most structures in the KLIFS annotation contain a ligand. The observation of structures in the DFG-out and DFG-in conformations reflect a sequence's ability to take on that conformation in the presence of a ligand when crystallized. A sequence with a high penalty for the DFG-out state will be unable to take on that conformation even in the presence of a type-II ligand.

We use PCA analysis of the structures based on 351 atom-atom pair distances which may be related to the DFG-in to DFG-out transition. When projected onto the first two principal components the structures form three clusters (figure S2). Many sequences classified as DFG-in by the KLIFS database are in an inactive Src-like conformation[4] in a cluster with PCA1$>$ 30. We limit our analysis to structures with PCA1$<$ 30, although we find using the full dataset does not qualitatively change our results. The DFG-in and DFG-out structures in the regions bounded by dashed boxes are used to calculate the DFG-out penalty score.

Contact Scores

A number of different methods have been suggested for obtaining a position-pair interaction score from the Potts model, including the "Direct information" [5], Frobenius norm [6], and APC-corrected Frobenius norm [7]. These methods account in different ways for the degeneracy of the Potts model parameters. As described in [3], while there are $\binom{L}{2}q^2$ bivariate and Lq univariate marginals only $\binom{L}{2}(q-1)^2 + L(q-1)$ of these are independent, with corresponding "gauge freedoms" in the J. One way to account for the degeneracy is to choose a particular gauge. Sets of parameters in one gauge may be transformed to another gauge by a certain set of gauge transformations while keeping all sequence probabilities fixed, for example adding or subtracting a constant from all the parameters.

We score interactions using a weighted Frobenius norm (see figure 1 in the main text), in which we first transform to a gauge which satisfies $\sum_\alpha w^{ij}_{\alpha\beta} J^{ij}_{\alpha\beta} = 0$, and then compute the score $\sqrt{\sum_{\alpha\beta}(w^{ij}_{\alpha\beta} J^{ij}_{\alpha\beta})^2}$. In the case the weights $w^{ij}_{\alpha\beta} = 1$ this reproduces an unweighted Frobenius norm calculation. For the purpose of contact prediction we use $w^{ij}_{\alpha\beta} = f^{ij}_{\alpha\beta}$, which, like the unweighted Frobenius norm, will give a score of 0 for uncoupled positions, but also downweights the contribution of couplings corresponding to infrequently observed mutant pairs which have high sampling error. For the kinase model with this score, 94% of the top scored 200 position-pairs greater than 4 positions apart along the sequence are contacts in at least 20% of structures in our PDB dataset with a 8Å nearest atom-atom contact cutoff distance, and 84% of the top scored 200 pairs are observed contacts with a 6Å cutoff.

Details of PMF calculation

We perform the threaded calculations using the (fully constrained) "zero gauge", in which $\sum_\alpha J^{ij}_{\alpha\beta} = 0$, as this gauge has the property that uncoupled positions will have $J^{ij}_{\alpha\beta} = 0$, suggesting that coupling values in this gauge may be interpreted as pairwise interaction strengths, and the lack of weighting means that individual coupling values are less affected by the presence of rare mutants in the dataset.

The DFG-in conformations typically have slightly more contacts than the DFG-out state, and therefore sum over a larger number of couplings. Since most couplings in evolved sequences are negative, this means the DFG-in state will have a lower threaded energy than the DFG-out state purely due to the different number of contacts. However the individual couplings used to compute DFG-in threaded energies are not significantly lower on average than the average coupling in the DFG-out threaded energies.

INVERSE ISING INFERENCE

The Inverse Ising inference procedure we use to infer the Potts model parameters proceeds by Markov Chain Monte Carlo (MCMC) sampling of sequences from a trial Potts Hamiltonian to obtain trial marginals, followed by a quasi-Newton parameter update step.

MCMC sampling

Following [8], we estimate marginals by generating sequences through MCMC on the Potts Hamiltonian for a given trial set of couplings. This method is mainly limited by sampling error and by the need for the simulation to reach equilibrium. We perform the computation on GPUs. Each work-unit of the GPU performs a MCMC walk, which proceeds by random point mutations to the protein sequence. As an optimization for the GPU all work units mutate the same random position simultaneously, but the mutant residue identity is computed independently.

The GPU gives an appreciable speedup over CPU. For our problem a Nvidia GeForce GTX Titan X GPU evaluates 1.4×10^8 MC steps per second for the $L = 175$, $q = 8$ system. In comparison, an 8-core 3.40GHz Intel Core i7-3770 CPU evaluates 3.6×10^6 steps per second with a nearly identical implementation. We ultimately run the inference using 4 GPUs in parallel.

Quasi-Newton optimization

We seek the set of fields h and couplings J which reproduce the data marginals f^{target} after sampling the model marginals f by MCMC. In [8], a quasi-Newton approach was developed in which a step direction in J and h was determined by inverting the system's Jacobian. The expected change in marginals Δf due to a change in J and h is given to first order by

$$\Delta f^{ij}_{\alpha\beta} = \sum_{xyab} \frac{\partial f^{ij}_{\alpha\beta}}{\partial J^{xy}_{ab}} \Delta J^{xy}_{ab} + \sum_{xa} \frac{\partial f^{ij}_{\alpha\beta}}{\partial h^{x}_{a}} \Delta h^{x}_{a} \qquad (2)$$

with a similar relation for Δf^{i}_{α}. By computing the Jacobian $\frac{\partial f^{ij}_{\alpha\beta}}{\partial J^{xy}_{ab}}$ and inverting the linear system of equation 2, we can solve for the step ΔJ and Δh which would give a desired Δf chosen to minimize the difference between model and data bivariate marginals. We choose $\Delta f = \gamma(f^{\text{target}} - f)$ and the damping factor γ is chosen small enough for the linear approximation to be valid.

A complication is that the f are not all independent due to "gauge freedoms", as described above. Because of this, equation 2 is noninvertible. However we may still solve the (nonindependent) linear system for any of its non-unique solutions, which will still produce the desired change Δf. Furthermore, using a nonindependent set of parameters allows simplification of the problem: We transform to a "fieldless" gauge in which $h^{i}_{\alpha} = 0$, as the remaining $\binom{L}{2}q^2$ couplings span the solution space. We only fit the bivariate marginals, which fully determine the univariate marginals.

In a fieldless gauge, we then seek to solve the simplified problem

$$\Delta f^{ij}_{XY} = \sum_{kl\alpha\beta} \frac{\partial f^{ij}_{XY}}{\partial J^{kl}_{\alpha\beta}} \Delta J^{kl}_{\alpha\beta}. \qquad (3)$$

The Jacobian is given by

$$\frac{\partial f^{ij}_{XY}}{\partial J^{kl}_{\alpha\beta}} = -f^{ijkl}_{XY\alpha\beta} + f^{ij}_{XY} f^{kl}_{\alpha\beta}. \qquad (4)$$

where $f^{ijkl}_{XY\alpha\beta}$ is a 4th-order marginal, which reduces to lower order marginals in the cases where the upper indices are equal to each other, and equals 0 in the case that two upper indices are equal but the corresponding lower indices are different. Solving equation 3 is challenging as the Jacobian is an $\binom{L}{2}q^2$ by $\binom{L}{2}q^2$ matrix. For L=200, q=8, typical of the problems we wish to solve, the Jacobian has over 10^{12} elements and is too large to store in computer memory. Following [8] we seek approximations to the linear system.

Self-Terms only

In [8], it is was assumed that each $f^{ij}_{\alpha\beta}$ only depends on the corresponding $J^{ij}_{\alpha\beta}$. That is

$$\Delta f^{ij}_{\alpha\beta} = \frac{\partial f^{ij}_{\alpha\beta}}{\partial J^{ij}_{\alpha\beta}} \Delta J^{ij}_{\alpha\beta} = (-f^{ij}_{\alpha\beta} + f^{ij}_{\alpha\beta} f^{ij}_{\alpha\beta}) \Delta J^{ij}_{\alpha\beta}. \qquad (5)$$

This is trivially inverted to give

$$\Delta J^{ij}_{\alpha\beta} = -\frac{\Delta f^{ij}_{\alpha\beta}}{f^{ij}_{\alpha\beta}(1 - f^{ij}_{\alpha\beta})}. \qquad (6)$$

Independent Pairs

A relaxed assumption is that each pair of positions is independent of other positions but each marginal depends on all the couplings at the same positions, that is, each f^{ij}_{XY} depends on $J^{ij}_{\alpha\beta}$ for all α, β. This is equivalent to a pair (L=2) system only, and in this section* we drop the i, j indices. In this pair system there are $q^2 - 1$ independent marginals (ie, all but one of the bivariate marginals, subject only to $\sum f_{\alpha\beta} = 1$), and in the fieldless gauge there are q^2 couplings, and thus only one gauge freedom. We seek to invert

$$\Delta f_{XY} = \sum_{\alpha\beta} \frac{\partial f_{XY}}{\partial J_{\alpha\beta}} \Delta J_{\alpha\beta}. \qquad (7)$$

Substituting equation 4 and dividing by f_{XY}, this can be rewritten as

$$(-\bar{I} + \bar{F})\overrightarrow{dJ} = \overrightarrow{df/f} \qquad (8)$$

where $\overrightarrow{df/f}$ is a vector with components $\frac{\Delta f_{XY}}{f_{XY}}$, $\bar{I}$ is the identity matrix and $\bar{F}$ is a matrix whose rows are the bivariate marginals. By rearranging and seeking an iterative solution, one finds this is solved (up to a constant due to the gauge freedom) by

$$\Delta J_{\alpha\beta} = -\frac{\Delta f_{\alpha\beta}}{f_{\alpha\beta}}. \qquad (9)$$

Perturbed Marginals

The marginals required in the update step of equation 9 must be determined from a computationally demanding MCMC sequence-generation run, but using a perturbative approach we evaluate the marginals for small changes in couplings without regenerating a new set of sequences, allowing many more approximate coupling update steps per round of MCMC.

When sampling N sequences for a set of couplings J, we expect to generate $n_s \sim e^{-E_s}$ sequences of type s. If we perturb the couplings to a new set J', we would

expect $n'_s \sim e^{-E'_s}$. We can simulate the effect of perturbing the marginals (without regenerating any sequences) by weighting the original set of sequences by a weight $w_s = e^{-(E'_s - E_s)}$, giving $n'_s = w_s n_s$. As $N \to \infty$ this approximation becomes exact. We calculate perturbed marginals $\tilde{f}$ as

$$\tilde{f}^{ij}_{XY} = \frac{1}{\tilde{N}} \sum_{s \in S_d} \delta^{s_i s_j}_{XY} e^{-(E'_s - E_s)} \qquad \tilde{N} = \sum_{s \in S_d} e^{-(E'_s - E_s)} \tag{10}$$

where s runs over our sequence dataset S_d. We find this approximation works quite well and is very quick to compute.

The accuracy of the approximation decreases as the coupling perturbation is increased because the overlap between the previously sampled sequences and "true" sequence distribution becomes small. In practice we limit the number of coupling update steps per MCMC round to a number chosen heuristically such that the bivariate marginals from the regenerated sequences are not too dissimilar from the prediction of the perturbed calculation.

Damping

The coupling update step consists of repeated calculation of weighted marginals followed by small updates to the couplings J. If the change in J per step becomes too large the updated set of couplings may take the system far from its previous position, preventing smooth progress towards the optimal solution. We account for this in part through the parameter γ described above, which we dynamically update. Additionally, to avoid divergent step sizes (ie, to avoid division by zero in equation 9 if $f = 0$) we use a modified step direction by adding flat pseudocount f_{pc} to the marginals to get pseudocounted marginals $\bar{f}$, with

$$\bar{f} = \frac{f + f_{\text{pc}}}{1 + f_{\text{pc}} q^2}. \tag{11}$$

Using these pseudocounted marginals in equation 9 we obtain a modified step direction

$$\Delta \bar{J}_{\alpha\beta} = -\frac{\Delta \bar{f}_{\alpha\beta}}{\bar{f}_{\alpha\beta}} = -\frac{\Delta f_{\alpha\beta}}{f_{\alpha\beta} + f_{\text{pc}}}. \tag{12}$$

The optimized solution for J will be independent of f_{pc} since at the solution $\Delta \bar{f}_{\alpha\beta} = \Delta f_{\alpha\beta} = 0$, and the pseudocount can be viewed as a damping factor. This pseudocount damping decreases the relative step size for couplings corresponding to small marginals where divergence is more likely, at the expense of increasing the number of necessary steps.

We find that it is useful to use a high value for f_{pc} such as 0.1 when the system is far from the solution, and as the system approaches the optimal solution (and the typical step sizes becomes smaller) f_{pc} can be decreased.

Inference Procedure

To perform the inference, we initialize J to values corresponding to an "independent" model where $h^i_\alpha = -\log f^i_\alpha$ and $J^{ij}_{\alpha\beta} = 0$ (and transform the the fieldless gauge), and choose an initial random sequence S_0. In each round of MCMC sequence generation we generate a set of sequences given the couplings J by running up to 131072 threads in parallel on the GPU, where each thread is an independent MCMC run starting from S_0. We equilibrate for a burn-in period of roughly 10^6 to 10^7 steps, and then collect samples of sequences at fixed intervals of MC steps, thus performing both a time and ensemble average. For the kinase inference we take 64 samples at intervals of roughly 10^5 steps, producing a total sequence set of up to 8 million sequences. Based on this sequence set we perform 64 perturbed coupling update steps using equation 12 with γ initialized to a value γ_0. If the bivariate marginal sum of squared residuals increases in any coupling update step, we halve γ and repeat the step, and otherwise double γ every 16 steps. We then assign a random sequence from the sequence sample to S_0 and repeat.

For the kinase inference, we perform three sequential inference rounds with different parameter values. We first minimize with $f_{\text{pc}} = 0.1$ for 15 rounds of MCMC generation with 16384 GPU threads equilibrated for 2.8×10^6 MC steps, followed by 15 rounds with $f_{\text{pc}} = 0.01$, each with 32768 GPU threads and 5.7×10^6 MC steps of equilibration, and finally run 30 rounds with $f_{\text{pc}} = 0.001$, each with 131072 GPU threads with 6.4×10^6 MC steps of equilibration. In all cases $\gamma_0 = 0.004$ and the inter-sample time is chosen such that the total samping period is equal to the equilibration period.

[1] M. Remmert, A. Biegert, A. Hauser, and J. Soding, Nat Meth **9**, 173 (2012).

[2] C. Feinauer, M. J. Skwark, A. Pagnani, and E. Aurell, PLoS Comput Biol **10**, e1003847EP (2014).

[3] F. Morcos, A. Pagnani, B. Lunt, A. Bertolino, D. S. Marks, C. Sander, R. Zecchina, J. N. Onuchic, T. Hwa, and M. Weigt, Proceedings of the National Academy of Sciences **108**, E1293 (2011).

[4] R. S. K. Vijayan, P. He, V. Modi, K. C. Duong-Ly, H. Ma, J. R. Peterson, R. L. Dunbrack, and R. M. Levy, J. Med. Chem. **58**, 466 (2015).

[5] M. Weigt, R. A. White, H. Szurmant, J. A. Hoch, and T. Hwa, PNAS **106**, 67 (2009).

[6] C. Lvkvist, Y. Lan, M. Weigt, E. Aurell, and M. Ekeberg, PRE **87**, 012707 (2013).

[7] L. Sutto, S. Marsili, A. Valencia, and F. L. Gervasio, Proceedings of the National Academy of Sciences **112**, 13567 (2015).

[8] A. Ferguson, J. Mann, S. Omarjee, T. Ndungu, B. Walker, and A. Chakraborty, Immunity **38**, 606 (2013).

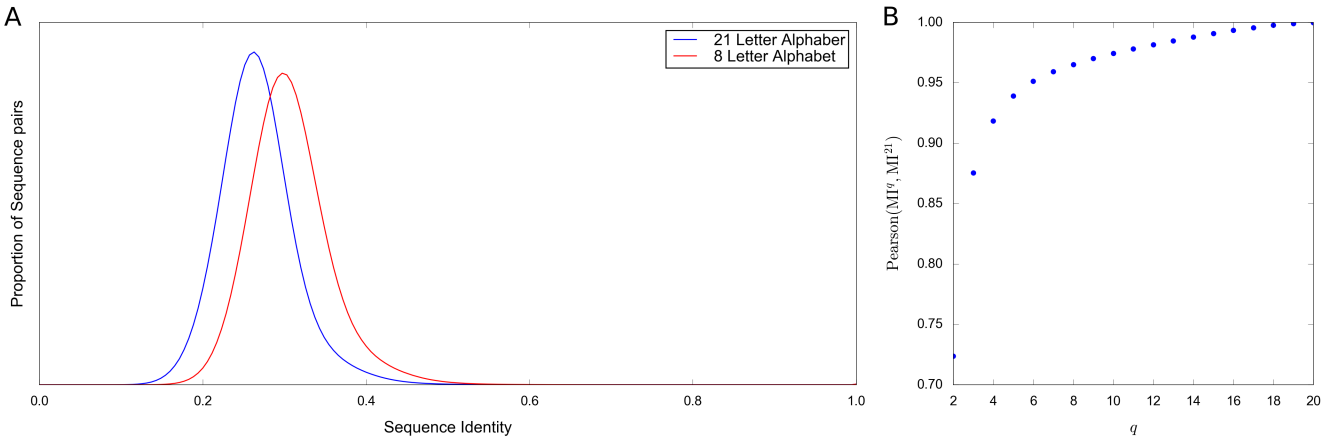

FIG. S1. Distribution of sequence identity scores (normalized inverse Hamming distance) between all pairs of sequences in the kinase dataset, computed for the original sequences using a 21 letter alphabet of 20 residues plus gap with phylogenetic weighting, and for the reduced 8 letter alphabet. The mean sequence identity is 27% for 21 letters and 31% for 8 letters.

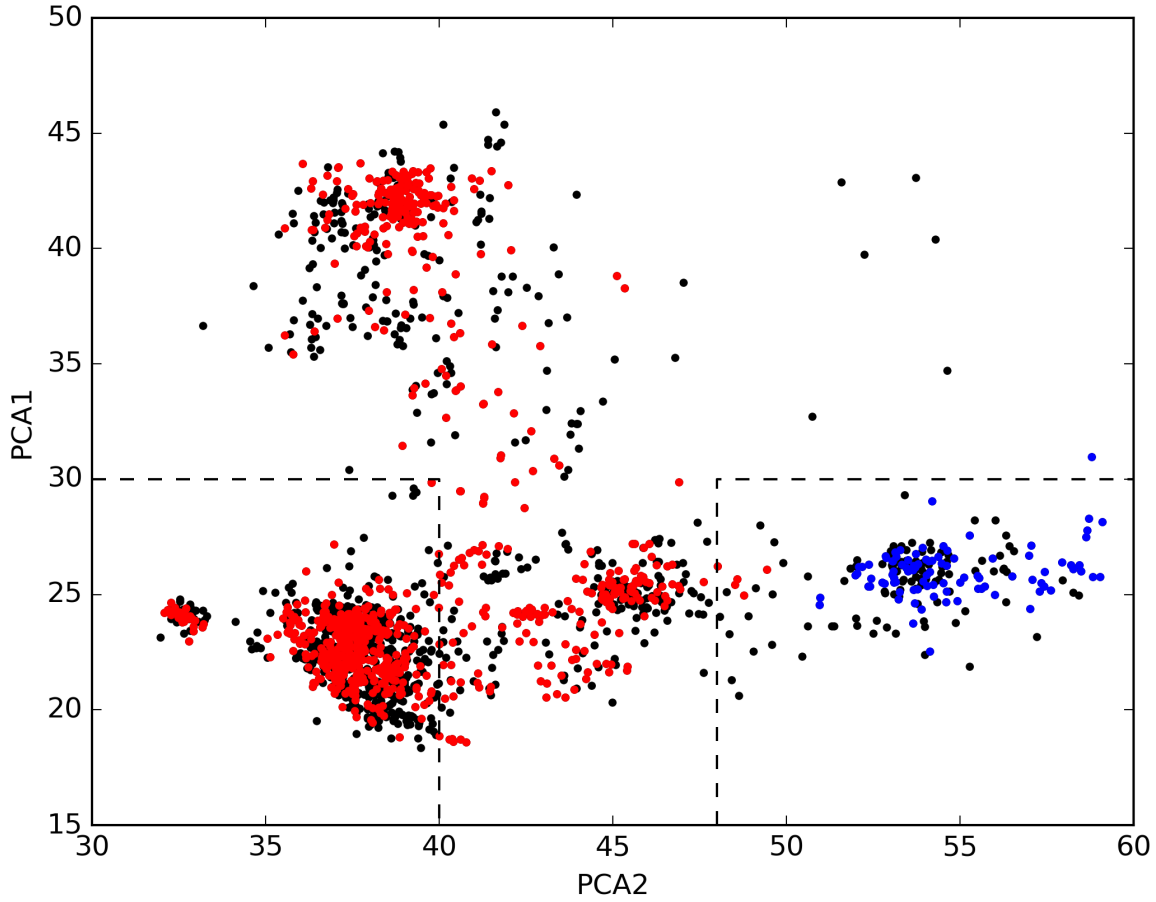

FIG. S2. PCA analysis of PDB structures, showing all structures (black) as well as structures annotated in the KLIFS database as DFG-in (red) and DFG-out (blue). Many sequences classified as DFG-in in the KLIFS database are in an inactive Src-like conformation (upper cluster).The DFG-in and DFG-out structures in the regions bounded by dashed boxes are used to calculate the DFG-out penalty score.